

Xilinx PROM Programmer

Parallel port serial PROM burner

By Müller & Müller

The programmer described here reads and writes to the Xilinx family of serial PROMs. The PC software is written in Turbo Pascal and the programmer plugs into a free printer (Centronics) port.

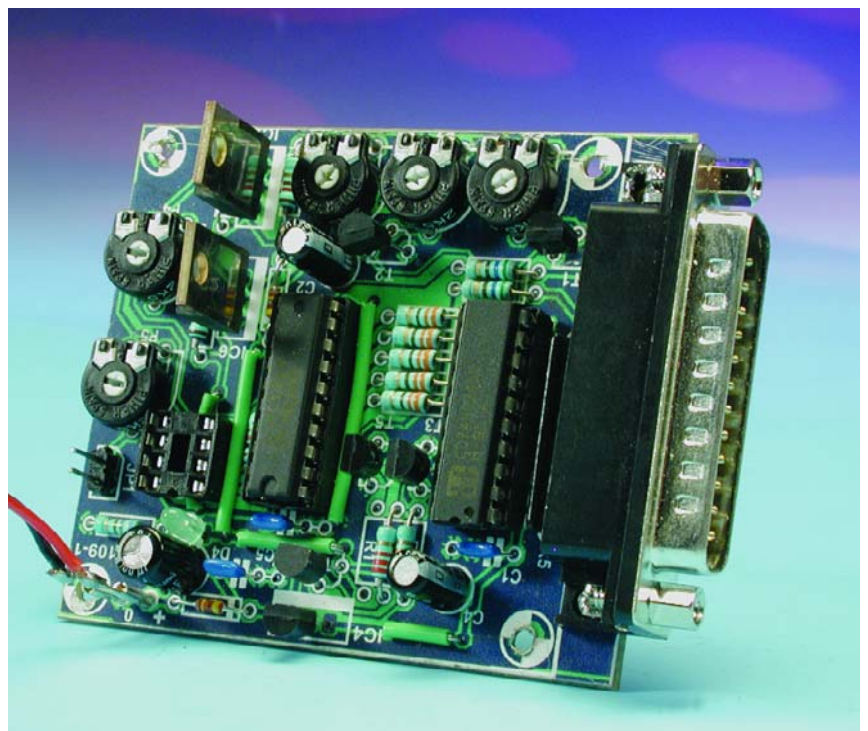
When it comes to PROM burning tools the most common model available is the universal type of PROM programmer. These are capable of handling the majority of chips currently on the market. The cost of such a burner is relatively high so the average electronics enthusiast has a hard job justifying his/her purchase especially if the programmer will only be used occasionally. If you do manage to scrape the funds to buy a programmer, regular software updates are fairly painless but it can be frustrating to find that a whole new model of programmer is introduced shortly after you make your purchase.

One solution to the problem is to build a low cost programmer and write (and update) the necessary programming software yourself. A disadvantage of choosing this path is that the programmer will not be so 'universal' as a commercial unit. The programmer described in this article is designed to burn serial PROMs produced by Xilinx but it should also be possible to adapt it for other manufacturers devices. These serial PROMs are used in conjunction with FPGAs (Field Programmable Gate Arrays). At boot-up the serial PROM loads configuration data to the FPGA. Together these devices are typically used to implement large complex digital systems (e.g., PCI bus designs).

The PROM burner described here plugs into a parallel printer port on a PC and the programmer software is written in Turbo Pascal and runs under DOS so even aged PCs should have no problem using it.

Burning Serial PROMs

The circuit diagram and PCB layout (Figure 3) are designed to accept PROMs with DIP out-



lines. For a more complete description of the devices refer to [1]. **Figure 1** shows the timing waveform specification for programming the XC17VXX-Family of PROMs.

As the name suggests, data to and from the PROM is transferred serially through a single pin (DATA) of the PROM. Writing begins at address '0000' with the data presented not in bits but in words. Most Xilinx PROMs have a 64-bit word length.

In order to switch the PROM into reading or writing mode it is neces-

sary to configure it with the correct bit pattern. These patterns are described under the heading *Programming* but for more detailed information see reference [2]. The timing diagram indicates the pattern and sequence of signals necessary for programming the PROM. Data is shifted into a 64 bit long latch in the PROM on the rising edge of the clock when both $\overline{CE}$ and $\overline{OE}$ are high. Once all 64 bits have been shifted into the register the voltage on pin V_{pp} is raised to V_{pp1} and the data is

latched. To read data from the PROM $\overline{CE}$ is now pulled Low and $\overline{OE}$ High. Data is shifted out of the PROM on every rising edge of the clock. Address incrementing occurs via the RESET/ $\overline{OE}$ pin.

In addition to simple data transfers to the PROM there are also special features that allow the RESET polarity to be inverted and also some internal information to be read-out, for example the PROM type and manufacturers code (MfG code).

The Hardware

The programmer circuit diagram is shown in **Figure 2**. Printer port outputs IN00 to IN07 are used along with the two inputs ACK (Acknowledge) and PE (Paper Empty). A standard printer extension lead connects a PC to the 25-way plug K1 on the programmer. Outputs from the PC are used to switch the voltages in the programmer and also to provide the data, clock and reset signal to the PROM. The two inputs to the PC carry the PROM output data and the status signal CEO.

Signals on the printer port from the PC are terminated by 4.7 k Ω pull-up resistors (R5) and buffered by the 74HC245 bus driver IC1. This IC carries the signals that switch voltages within the programmer so it is important to ensure that its supply voltage is derived from a separate (unswitched) 5 V regulator (IC4). Serial data for the PROM also passes through IC1 and IC3. The supply for IC3 comes from the switched V_{CC} . This ensures that voltage thresholds of the PROM signals correspond to V_{CC} and that the signals DATA, CLK, $\overline{OE}$ and $\overline{CE}$ (and the acknowledge signals ACK and PE) will only be driven when the supply voltage V_{CC} is switched on. To be absolutely sure that V_{CC} cannot be pulled high via the input protection diodes of IC3, transistor T5 switches the supply to ground via the 1 k Ω resistor when V_{CC} is switched off.

The programmer can handle both 3.3 V and 5 V PROMs so it's important to use an **HC type** of driver for IC3. The LS or ALS family of devices do not provide suitable output signal

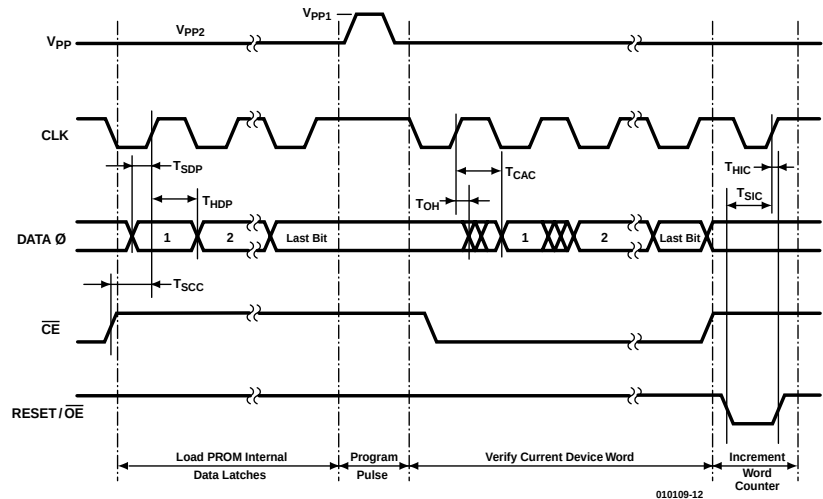


Figure 1. Programming cycle timing diagram for the XC17VXX type PROM.

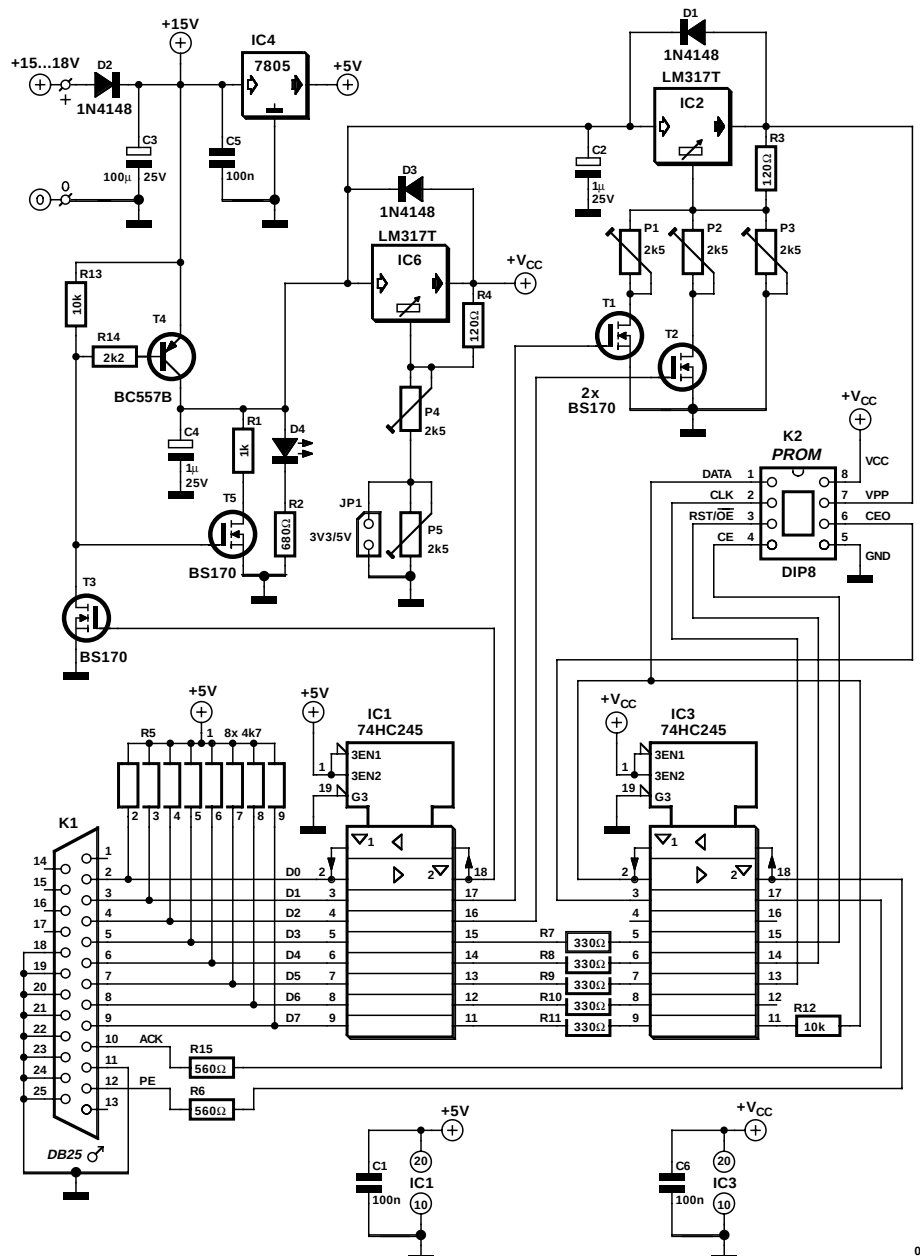


Figure 2. The circuit diagram.

COMPONENTS LIST

Resistors:

R1 = 1k Ω
 R2 = 680 Ω
 R3, R4 = 120 Ω
 R5 = 8-way 4k Ω 7 SIL array
 R6, R15 = 560 Ω
 R7-R11 = 330 Ω
 R12, R13 = 10k Ω
 R14 = 2k Ω
 P1-P5 = 2k Ω 5 preset

Capacitors:

C1, C5, C6 = 100nF
 C2, C4 = 1 μ F 25V radial
 C3 = 100 μ F 25V radial

Semiconductors:

D1, D2, D3 = 1N4148
 D4 = LED, red
 IC1, IC3 = 74HC245
 IC2, IC6 = LM317T
 IC4 = 7805
 T1, T2, T3, T5 = BS170
 T4 = BC557B

Miscellaneous:

JP1 = 2-way SIL header with jumper
 K1 = 25-way sub-D plug (male), PCB mount
 K2 = 8-way DIL socket with turned pins
 PCB, order code **010109-1**, available from
 The PCBShop
 Disk, project software, order code **010109-11** or Free Download

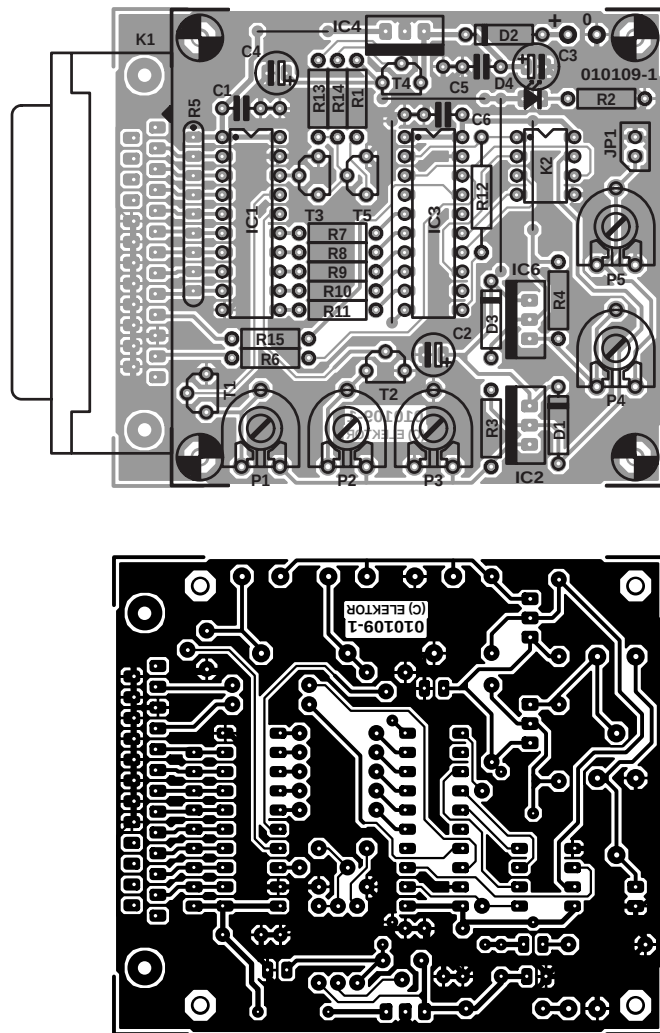


Figure 3. The Xilinx programmer uses a single sided PCB.

levels and the HCT family cannot operate at the necessary supply voltages. HC devices can operate with a supply of between 2 V and 6 V. The series resistors in the data inputs of IC3 limit the input current when $V_{cc} < 5$ V.

The DATA pin of the PROM can be configured as either an input or output. Resistor R12 is used to effectively increase the output impedance of IC3 pin 11, this ensures that whenever the PROM drives data out from pin 1 it will always be available as an input to the PC via line PE irrespective of the state of the output of IC3 pin 11. It is only when writing to the PROM (when the DATA pin will be an input and therefore high impedance) that the output of IC3 pin 11 (write data) will appear on the DATA pin. An important parameter of the data transfer is the slope of the signal edges of the PE connection to the PC. Resistor R6 is used to reduce the effects of reflections in the cable that may otherwise corrupt the signal.

The circuit requires a single external input voltage in the range of 15 to 18 V; all other voltages required for programming are

derived on-board. To be on the safe side all PROM voltages are switched by signals via the printer port.

The programmer can use PROMs with either a 5 V or 3.3 V supply voltage (V_{cc}), jumper JP1 selects between these two supply levels and gives a simple visual indication of the supply voltage selected: with JP1 fitted the supply voltage is 3.3 V, with JP1 not fitted the supply voltage will be 5 V. Pin 7 (V_{pp}) requires three different voltage levels depending on the operation that the PROM is performing: V_{pp1} for programming, V_{pp-nom} for reading the data and V_{pp2} for verification of the writing procedure. Transistors T1 and T2 switch these three voltage levels from data signals D1 and D2. With V_{cc} switched on and D1 and D2 low the level on this pin will be V_{pp1} . The programmer voltage sequencing

is given in the software. The program also indicates the correct voltage levels necessary depending on the type of PROM selected so it's not necessary to wade through piles of data sheets for this information. The LED indicates that reading and programming voltages are available.

Software features

The programmer software is written in Turbo Pascal and runs under DOS using command line options. Among the software downloads for this article a free version of DOS is included, this enables a boot disk to be made so that the programmer software will run even if you do not have DOS available on your machine. Using the command **spb /?** gives an overview of all the available options such as selection of chip type and

the type of access required of the PROM (eg reading, writing or reset bit reading etc). The program always asks if the chip is plugged into the socket before it attempts to switch the supply voltage to the PROM. The program checks if the PROM in the socket corresponds with type of chip selected and also documents the programming process.

The program options are described below in the same order that they are used in the program:

/?

Overview of the program options.

/d?

The printer port where '?' is the number of the LPT port connected to the programmer.

/av

This command is used to setup the different programming voltages required for the specified PROM. An oscilloscope or DVM can be used to test voltage levels. The program indicates the required voltage and on which pin of the PROM socket the voltage should be measured. It also indicates which preset needs to be adjusted. The chip name is only used here for the sake of appearance.

/m

This function reads the chips MfG code. A chip name must be entered here. When the MfG code is correct then the message Chip identified will be displayed; if not the MfG code read from the chip will be displayed. If it is necessary to enter a new chip into the chip information file (chip-info.dat), all the displayed values must first be converted to hexadecimal.

All further functions are specific to each chip type and will only be carried out if the MfG code read from the chip corresponds to the selected chip type. The MfG code is always tested first.

/rp

Reads the polarity of the reset bit.

/bc

Blank Check, this tests if all the PROM bits are blank (= 1). The programming specification suggests that only a few bits need to be

checked to determine if the chip is blank but to be sure the program performs a blank check on the entire PROM, the extra time needed is not excessively long.

/b

Burn, the data held in the brenndat.bin file is programmed into the PROM, this operation will not program the polarity of the reset signal. The file name is stored as the constant BrenndatenFileName and can be changed in the program at this point. The size of the data in the file must always be smaller or equal to the available space in the PROM. If the file size is bigger than the PROM size given in 'Size' an error message is generated and the burning process will not start.

/r

This command reads all the data in the PROM and stores it in the file data.bin. This feature will execute differently depending on whether the reset bit has already been programmed. The reset polarity should always be read before this command is used.

/cp

Changes the reset polarity. First the polarity of the reset is tested using /rp and if necessary the reset polarity is changed. Lastly the reset status will be read and displayed.

Two examples

In order to read a XC17S10 type of PROM connected to port LPT1 the following command line is used: spb /r s10 /d1

Where s10 specifies the PROM type. Using the command spb /av /cp /b s200a /d2 indicates that the PROM type has changed and the programming voltages will be set up and adjusted before the PROM is inserted into the socket. The reset polarity will then be changed and Data will be programmed into the XC17S200A via the LPT2 printer port.

In and out

In order to be able to switch all the voltages and signals in the programmer hardware each bit of the printer port byte must be individually programmable. In the software the state

of the eight printer port output bits is held in the eight bit wide global variable dat. As an example if we wanted to switch on the supply voltage to the programmer we would need to set bit D0 (connector K1 pin 2) to a 1. This could be accomplished with the line:

```
dat := dat or $01
```

This will set the bit in software and then with an output instruction to the printer port this new value of D0 will be output to the programmer leaving bits D1 to D7 unchanged. To turn off the supply voltage we just need to do the reverse procedure:

```
dat := dat and $fe
```

or alternatively:

```
dat := dat and (not $01)
```

And again write it to the printer port. The hexadecimal value of each bit of the LPT port (and its inverse hex value) is given below:

\$01 (\$FE)	Vpp - Relay (REL)
\$02 (\$FD)	Vpp-nom
\$04 (\$FB)	Vpp2
\$08 (\$F7)	$\overline{CE}$
\$10 (\$EF)	RESET/ $\overline{OE}$
\$20 (\$DF)	CLK
\$80 (\$7F)	DATA

Chipinfo.dat

All the data relating to the chip are stored in this file. The information required by the programmer is:

Mm This byte in the identification code identifies the chip manufacturer (Xilinx = \$C9).

IDCLK This is the number of clock pulses needed to read out the manufacturers code.

ID is the second part of the MfG code indicating the chip series and type.

Size is the size of the PROM in bits.

RSTClk This is the number of clock pulses needed to program the Reset bit.

Vcc gives the PROM operating voltage.

Vpp1 gives the PROM programming voltage.

Vpp2 This voltage on pin 7 used to test/fix the data written into the PROM. This voltage level is only slightly higher than the supply voltage and is not implemented at present since verification at this slightly raised voltage is not essential.

Vppnom The voltage on the pin V_{PP} during reading. It will be at least as high as the chips supply voltage V_{CC}.

Bugfix

The program is written in Turbo Pascal and uses a system unit called *Crt*, this unit includes a bug which causes a run time error in Windows if the PC's processor is a Pentium type running faster than about 200MHz. The program runs without problem in DOS.

A patch is included in the software package **010109-11** for this article. The patch program (TPPATCH.EXE) is able to locate and correct the failure in this unit. If you need to make any changes to the source code for this project it is therefore important to run this patch program once you have recompiled *spb.exe*.

Name This is an abbreviated name identifying the type of PROM used in the command line to call the program. The same name must not be used for PROMs from other manufacturers even if the PROM has the same memory capacity.

The four voltages V_{CC} , V_{pp1} , V_{pp2} and V_{ppnom} are used to simplify the voltage settings when different types of PROMs are used. The levels are displayed by the */av* function. The file can be expanded to incorporate other serial PROMs that have compatible programming algorithms. The maximum number of entries is given by the *maxdatas* constant defined in *spb.pas* and will need to be changed if new devices are added. All the data in *chipinfo.dat* is taken from [2].

Reading the MfG code

Each PROM contains its manufacturer's identification code that enables it to be recognised by the programmer. The code consists of two bytes: the first byte identifies the manufacturer (Xilinx uses the code \$C9). The second byte comprises four bits indicating the density of the PROM and four bits for information on the programming algorithm. This is contained in the second column of *chipinfo.dat*.

The PROM's manufacturer code is read in the *ReadMfGCode* procedure. The PROM must first be set to program mode with $\overline{CE} = 1$ and $\overline{OE} = 0$. Next a defined number of clock pulses is sent to the PROM these are described as the N_{IDCLK} or also the *IDCLK* (see the first column of the *chipinfo.dat*). Now with $\overline{OE} = 1$ and $\overline{CE} = 0$ the MSB of the MfG-Code is output on the *DATA* pin. The code is now read by the *LeseByteMfG* function as each bit is shifted out by the clock signal.

The Reset Bit

The polarity of the serial PROMs reset signal can be programmed. When the PROM is used together with a Xilinx FPGA it is useful to be able to reset the PROM from the INIT signal from the FPLA. This means that the polarity of the PROM reset pin needs to be changed so that $RESET = 0$. Changing the polarity of the reset signal is a one-way operation so once the polarity has been programmed to negative reset it cannot be changed back. For trials it is best to test the chip with an unprogrammed reset ($RESET = 1$).

Programming

The Burning procedure switches the PROM into programming mode. Both $\overline{CE}$ and $\overline{OE}$ must be high for at least two positive clock edges and the voltage on V_{pp} must be at V_{pp1} [1]. In this mode the manufacturer and chip information can be read and also data can be written to the PROM. The data is shifted serially into the PROM in 64 bit chunks and latched by a V_{pp1} pulse on pin V_{pp} . The XC17S05D Xilinx PROM is the only type with 32 bit long latches; it is not widely used and is not supported by this programmer. According to the specification after programming the voltage level on V_{pp} should be set to V_{pp2} but this is not implemented in the program. Lastly the programmer switches out of program mode and both supply voltages V_{CC} and V_{pp} are switched off.

Build and operation

The single-sided PCB has a board area of 72.63 mm², this gives enough space for the components to be mounted quite easily. The five wire links need to be fitted first (one of these runs under the 8 pin DIP programming socket).

Once you are sure that all the polarised components have been fitted the correct way round (including the SIL resistor network) and double-checked your soldering the circuit can be powered up. The first test should take place without a PROM fitted to the programming socket. Connect 15 V to 18 V between the supply solder pins on the PCB. Diode

D2 ensures that no damage occurs if you accidentally mix up the power supply polarity. First check that 5 V is available on pin 20 of IC1, LED D4 should also light to indicate that the supply voltage is available.

The software can now be installed and run to drive the programmer. Jumper JP1 allows the supply voltage of the PROM to be quickly changed between 5 V and 3.3 V depending on the type of PROM used. This supply can also be adjusted to suit PROMs using other supply voltages. Xilinx are currently offering three PROMs families each with a different supply voltage. The supply voltage data for the PROMs is stored in the *Chipinfo.dat* file and can be read using the */av* option so the voltage levels can be quickly adjusted.

After installation the PCB can be plugged into the printer socket of the PC. The LED will go out and commands can be entered to the programmer software. For example using the command line *spb /av /d1 s10* calls the program where */av* indicates the voltage test and */d1* is for printer port LPT1 (*d2* would be entered here if LPT2 was used) and *s10* indicates the PROM type.

Next it is necessary to adjust the two-level supply voltage to the PROM. Firstly fit jumper JP1 and with a DVM connected to the output of IC6 (V_{CC}) adjust the preset P4 until V_{CC} reaches 3.3 V, once this has been achieved remove JP1 and adjust preset P5 until V_{CC} is 5 V.

Finally the program indicates adjustment of the programming voltages on pin 7. These settings are best made when you need to change to a different type of PROM.

(010109-1)

References:

- [1] Programmer Specification
www.xilinx.com/support/programr/specs.htm
- [2] PROM Specification
www.xilinx.com/isp/csapps.htm